

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
Before the Board of Patent Appeals and Interferences

an re Patent Application of

DIJKSTRA

Serial No. 10/633,362

Filed: August 4, 2003

Title: ADDRESS GENERATION



Atty Dkt. SCS-550-455

C# M#

TC/A.U.: 2181

Examiner: J. Moll

Date: May 30, 2007

AFG
SAB

Mail Stop Appeal Brief - Patents

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

☐ **Correspondence Address Indication Form Attached.**

☐ **NOTICE OF APPEAL**

Applicant hereby **appeals** to the Board of Patent Appeals and Interferences

from the last decision of the Examiner twice/finally rejecting \$500.00 (1401)/\$250.00 (2401) \$
applicant's claim(s).

☒ An appeal **BRIEF** is attached in the pending appeal of the
above-identified application \$500.00 (1402)/\$250.00 (2402) \$ 500.00

☐ Credit for fees paid in prior appeal without decision on merits -\$()

☐ A reply brief is attached. (no fee)

☐ Petition is hereby made to extend the current due date so as to cover the filing date of this
paper and attachment(s)
One Month Extension \$120.00 (1251)/\$60.00 (2251)
Two Month Extensions \$450.00 (1252)/\$225.00 (2252)
Three Month Extensions \$1020.00 (1253)/\$510.00 (2253)
Four Month Extensions \$1590.00 (1254)/\$795.00 (2254) \$

☐ "Small entity" statement attached.

Less month extension previously paid on -\$()

TOTAL FEE ENCLOSED \$ 500.00

Any future submission requiring an extension of time is hereby stated to include a petition for such time extension.
The Commissioner is hereby authorized to charge any deficiency, or credit any overpayment, in the fee(s) filed, or
asserted to be filed, or which should have been filed herewith (or with any paper hereafter filed in this application by this
firm) to our **Account No. 14-1140**. A duplicate copy of this sheet is attached.

901 North Glebe Road, 11th Floor
Arlington, Virginia 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100
SCS:kmm

NIXON & VANDERHYE P.C.
By Atty: Stanley C. Spooner, Reg. No. 27,393

Signature: _____



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Patent Application of

DIJKSTRA

Serial No. 10/633,362

Filed: August 4, 2003

For: ADDRESS GENERATION

Confirmation No.: 5128

Atty. Ref.: 550-455

Group: 2181

Examiner: J. Moll

APPEAL BRIEF

On Appeal From Group Art Unit 2181

Stanley C. Spooner
NIXON & VANDERHYE P.C.
11th Floor, 901 North Glebe Road
Arlington, Virginia 22203
(703) 816-4028
Attorney for Appellant

05/31/2007 JADD01 00000031 10633362
01 FC:1402 500.00 OP



TABLE OF CONTENTS

I. REAL PARTY IN INTEREST	1
II. RELATED APPEALS AND INTERFERENCES.....	1
III. STATUS OF CLAIMS	2
IV. STATUS OF AMENDMENTS.....	2
V. SUMMARY OF THE CLAIMED SUBJECT MATTER	2
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL.....	6
VII. ARGUMENT	6
A. The Examiner fails to properly construe the means-plus-function and step-plus-function claims.....	7
B. The Examiner fails to establish that the Applicant Admitted Prior Art (AAPA) discloses structures positively recited in independent apparatus claim 1 and independent method claim 22	10
C. The Examiner fails to provide any reason or motivation for combining different portions of the AAPA in the manner of Appellant's claims.....	13
D. The Examiner fails to identify where the steps of claim 22 are set out in AAPA.....	15
E. There is no support for the rejection of claims 1-42 under 35 USC §102 as anticipated by Applicant Admitted Prior Art	16
VIII. CONCLUSION.....	19
IX. CLAIMS APPENDIX	A1
X. EVIDENCE APPENDIX.....	A13
XI. RELATED PROCEEDINGS APPENDIX	A14



**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Patent Application of

DIJKSTRA

Atty. Ref.: 550-455

Serial No. 10/633,362

Group: 2181

Filed: August 4, 2003

Examiner: J. Moll

For: ADDRESS GENERATION

May 30, 2007

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

I. REAL PARTY IN INTEREST

The real party in interest in the above-identified appeal is ARM Limited by virtue of an assignment of rights from the inventor to ARM Limited recorded December 11, 2003 at Reel 14793, Frame 779.

II. RELATED APPEALS AND INTERFERENCES

There are believed to be no related appeals, interferences or judicial proceedings with respect to the present application, other than the Pre-Appeal Brief Request for Review previously filed in this appeal.

III. STATUS OF CLAIMS

Claims 1-42 stand rejected in the Final Official Action. The Examiner contends that with respect to claims 1-42, each of these claims is anticipated under 35 USC §102(a) by Applicant Admitted Prior Art (Figs. 1-5, Description of the Prior Art, pages 1-5).

The above rejection of claims 1-42 is appealed.

IV. STATUS OF AMENDMENTS

An Amendment under Rule 116 was filed on January 29, 2007, and was entered by the Examiner as noted on the Advisory Action mailed February 20, 2007 (Paper No. 20070220).

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

Appellant's specification and figures provide an explanation of the claimed invention set out in independent claims 1 and 22, with each claimed structure and step addressed as to its location in the specification and in the figures.

“1. A data processing apparatus comprising:

a processor core [item 10 shown in figure 1 and discussed on page 1, line 6 to page 2, line 13 and elsewhere in the specification] operable to process a sequence of instructions, said processor core having a plurality of pipeline stages

[first execution stage 80, second execution stage 90 as shown in figure 2 and discussed on page 1, line 22 to page 2, line 13 and elsewhere in the specification], one of said plurality of pipeline stages being an address generation stage [item 200 shown in figure 6A and discussed on page 13, line 6 to page 14, line 24 and elsewhere in the specification] operable to generate an address associated with an instruction for subsequent processing by said pipeline stages, said instruction being one from a first group of instructions or a second group of instructions, said address generation stage comprising:

address generation logic [item 220 shown in figures 6A & 6B and discussed on page 14, line 18 to page 16, line 26 and elsewhere in the specification] for receiving operands associated with said instruction, for generating a shifted operand from one of said operands [interconnect logic 260 as shown in Figure 6B and discussed on page 16, line 21 to page 17, line 4 and elsewhere in the specification], and for adding together, in dependence on said instruction, selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages; and

operand routing logic [items 205 and 215 shown in figure 6A and discussed on page 14, line 28 to page 16, line 14 and elsewhere in the specification], in dependence on said instruction, for routing operands associated with instructions from said first group of instructions to said address generation logic [220] and for routing operands associated with instructions from said second group of

instructions via operand manipulation logic [item 216 shown in figure 6A and discussed on page 14, line 18 to page 16, line 26 and elsewhere in the specification] for manipulation of said operands prior to routing to said address generation logic.”

“22. (previously presented) In a data processing apparatus comprising a processor core [item 10 shown in figure 1 and discussed on page 1, line 6 to page 2, line 13 and elsewhere in the specification] operable to process a sequence of instructions, said processor core having a plurality of pipeline stages [first execution stage 80, second execution stage 90 as shown in figure 2 and discussed on page 1, line 22 to page 2, line 13 and elsewhere in the specification], one of said plurality of pipeline stages being an address generation stage [item 200 shown in figure 6A and discussed on page 13, line 6 to page 14, line 24 and elsewhere in the specification] operable to generate an address associated with an instruction for subsequent processing by said pipeline stages, said instruction being one from a first group of instructions or a second group of instructions, a method of generating said address comprising the steps of:

a) receiving, at address generation logic, operands associated with said instruction [items 200 and 220 shown in figures 6A & 6B and discussed on page 14, line 18 to page 16, line 26 and elsewhere in the specification];

b) generating a shifted operand from one of said operands [interconnect logic 260 as well as items 200 and 220 shown in figures 6A & 6B and discussed on page 14, line 18 to page 17, line 4 and elsewhere in the specification];

c) adding together, in dependence on said instruction, selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages [items 200 and 220 shown in figures 6A & 6B and discussed on page 14, line 18 to page 16, line 26 and elsewhere in the specification];

d) routing, in dependence on said instruction, operands associated with instructions from said first group of instructions to said address generation logic [items 205 and 215 shown in figure 6A and discussed on page 14, line 28 to page 16, line 14 and elsewhere in the specification]; and

e) routing, in dependence on said instruction, operands associated with instructions from said second group of instructions via operand manipulation logic for manipulation of said operands prior to routing to said address generation logic [item 216 shown in figure 6A and discussed on page 14, line 18 to page 16, line 26 and elsewhere in the specification].”

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-42 stand rejected under 35 USC §102(a) as being anticipated by Applicant Admitted Prior Art (Figs. 1-5, Description of the Prior Art, pages 1-5) (AAPA).

VII. ARGUMENT

Appellant's arguments include the fact that the burden is on the Examiner to first and foremost properly construe the language of the claims to determine what structure and/or method steps are covered by that claim. After proper construction of the claim language, the burden is also on the Examiner to demonstrate where a single reference (in the case of anticipation) or a plurality of references (in the case of an obviousness rejection) teaches each of the structures and/or method steps recited in independent claims 1 and 22.

The Court of Appeals for the Federal Circuit has noted in the case of *Lindemann Maschinenfabrik GMBH v. American Hoist & Derrick*, 221 USPQ 481, 485 (Fed. Cir. 1984) that "[a]nticipation requires the presence in a single prior art reference disclosure of **each and every element** of the claimed invention, **arranged as in the claim**" (emphasis added).

A. The Examiner fails to properly construe the means-plus-function and step-plus-function claims

Appellant's claim 1 states address generation logic "for receiving," "for generating" and "for adding." No structure for accomplishing these three functional interrelationships is set out anywhere else in the claim. Claim 1 also recites operand routing logic "for routing operands associated with instructions from said first group . . . and "for routing operands associated with instructions from said second group via operand manipulation logic" Again, no structure is identified which will perform these claimed functions and claimed interrelationships. Federal circuit decisions in *Donaldson* and others confirm that if the claims are in "means" form or, if not using the word "means" but failing to provide structure for performing the claimed functions, "the claim shall be construed to cover the corresponding structure . . . in the specification and equivalents thereof." (35 USC §112(6th)).

Since the above noted functions are specified in the independent claims and no structure capable of performing that function is recited in the claim, the claim must be construed as a means-plus-function claim, notwithstanding an initial presumption against means-plus-function status if the word "means" is not used ("the presumption here is overcome and the phrase 'colorant selection mechanism' should be construed as a means-plus-function limitation." *The Massachusetts Institute of Technology v. Abacus Software*, 80 USPQ2d 1225, 1231 (Fed. Cir.

2006)). Therefore, the Examiner failed to properly consider independent claim 1 as being in “means” form.

The Examiner summarily dismisses the Applicant’s § 112 (6th paragraph) claim construction argument by stating that “limitations from the specification are not read into the claims.” Appellant does not contend that limitations in the specification must be read into the claims – if fact just the opposite, the limitations in the claims themselves must be found in any cited prior art. Moreover, the Examiner’s conclusion is clearly wrong in view of the statutory requirement of 35 USC § 112 (6th paragraph) and the subsequent decisions by the Federal Circuit in *In re Donaldson* and its progeny.

The Examiner alleges that the word “logic” as used in the claim means “computer circuitry” but fails to provide any evidence of record supporting his conclusion (while the Examiner cites the American Heritage Dictionary, he fails to make of record any copy of any portion of that dictionary (so that the context of his alleged definition can be considered) to support his conclusion introduced for the first time in the Final Rejection).

Appellant introduces in response a copy of page 1330 from Webster’s third New International Dictionary which defines “logic” as “interconnection or connection or sequence (as of facts or events) esp. when seen by rational analysis as inevitable, necessary or predictable” This definition of “logic” does not

necessarily mean “circuits” or any particular structure and supports the above noted consideration of claim 1 as being in “means” form.

Moreover, even if the Examiner is correct, assuming that logic means circuits and assuming that this defeats the “means” argument, that circuitry must be construed as stated in the claims, e.g., for “generating logic” it must receive “operands associated with said instruction,” generate “a shifted operand from one of said operands,” and add “together . . . selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages.” These are the limitations and interrelationships that the Examiner has not properly considered in the final rejection of the independent claims.

The Examiner must demonstrate where the AAPA discloses these positively recited claim limitations, such as the address generating logic’s “receiving operands associated with said instructions, generating a shifted operand from one of said operands, and for adding together in dependence on said instruction, selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages” and the operand routing logic’s variously recited “routing” functions. Where these elements exist in the AAPA has not been shown by the Examiner.

Moreover, even if the individual elements were somehow disclosed in the AAPA, the burden is also on the Examiner to show how and where the AAPA shows that these elements are “arranged as in the claim” as required by the above

noted *Lindemann* case. The Examiner fails to point to any disclosure of the interrelationship between the two logic elements in independent claim 1 as required by the claim

Without a correct analysis of independent claim 1, it is, of course, impossible for the Examiner to demonstrate how or where Applicant's Admitted Prior Art contains any disclosure of the properly construed and claimed structures or their claimed interrelationship.

B. The Examiner fails to establish that the Applicant Admitted Prior Art (AAPA) discloses structures positively recited in independent apparatus claim 1 and independent method claim 22

(1) No disclosure of "address generation logic"

Independent claim 1 positively recites "address generation logic . . . for generating a shifted operand from one of said operands." In the outstanding Final Rejection, the Examiner alleges (on page 4) that this claimed structure is disclosed in "ADD unit 170." There is no evidence of record provided by the Examiner to suggest that the disclosed ADD unit 170 can generate a "shifted operand" from one of the received operands. In fact, ADD unit 170 cannot possibly generate a shifted operand because it does not have any capability of generating a shifted operand. The Examiner has provided no evidence of record to establish that ADD unit 170 has any inherent capability of generating a shifted operand.

The Examiner attributes his rationale to the “Free On-line Dictionary of Computing,” but even that definition (also not of record in this case) does not help the Examiner’s case. There is no indication in that definition or anywhere else in the specification that multiplexer 130 (which is part of ADD unit 170) anticipates the features of Appellant’s claim 1, i.e., the address generation logic for “generating a shifted operand from one of said operands.” The multiplexer 130 in Figure 4 is capable of receiving a shifted operand as an input and merely outputs that same shifted operand, either directly or in inverted form.

There is no disclosure identified by the Examiner which would lead one of ordinary skill in the art to believe that either multiplexer 130 or ADD unit 170 can generate a “shifted operand.” The failure of the Examiner to provide any evidence to support his conclusion is evidence that the AAPA does not anticipate or render obvious the subject matter of independent claims 1 or 22.

(2) No disclosure of operand routing logic “for routing operands” as specified in claims 1 and 22

The Examiner suggests on page 5 of the Final Rejection that multiplexers 155 and 165 in Figure 5 are the equivalent to the claimed “operand routing logic.” He also contends that shifter 160 in Figure 4 is a counterpart of the operand manipulation logic recited in claim 1.

However, a reading of AAPA Figure 4 (and the accompanying specification discussion of Figure 4) will show that, if the Examiner contends the

shift circuitry 135 is a counterpart to the claimed operand manipulation logic and if the multiplexers 130 and 137 in Figure 4 are counterparts of the operand routing logic, then the shift circuitry 135 being considered a counterpart of the address generation logic is completely inconsistent. Thus, the operand routing logic as set out in claim 1 is clearly missing from the AAPA.

The AAPA arrangements in Figures 3A and 3B in the present specification disclose known arrangements which combine a shift and add operation. However, even if this previously known arrangement is considered to be a counterpart to the claimed "address generation logic," it could not be encompassed by claim 1 because there is no disclosed counterpart for the "operand routing logic."

As a result of the above, none of the AAPA arrangements discussed with respect to Figures 1-5 in the present application anticipate the subject matter of independent claims 1 and 22. In order to establish a *prima facie* case, the Examiner must pick and choose different claimed elements from the different embodiments disclosed, i.e., the embodiment of Figures 3A and 3B, the embodiment of Figure 4, the embodiment of Figure 5.

There is no precedent for a rejection under §102 to be based upon elements taken from different references. While the AAPA is extracted in the Background of the Invention portion of the specification, it is not included in a single prior art publication. As a result, it cannot support a rejection under §102 unless all claimed elements and all claimed interrelationships are in a single reference or

embodiment. As noted above, the Examiner is picking and choosing elements and interrelationships from three separate embodiments.

The burden is on the Examiner to identify a single disclosed embodiment which “anticipates” under §102 the subject matter of Appellant’s claims. There is no single disclosed embodiment (from the three AAPA embodiments of (1) Figures 3A and 3B, (2) Figures 4A and 4B, and (3) Figure 5) which contains all of the claimed elements and claimed interrelationships, and therefore there can be no *prima facie* case of anticipation.

Additionally, because the burden is on the Examiner to establish a *prima facie* case, and since he has failed to meet this burden, there is simply no support for a rejection of independent claims 1 and 22 under 35 USC §102 or §103.

C. The Examiner fails to provide any reason or motivation for combining different portions of the AAPA in the manner of Appellant’s claims

Because, as noted above, Appellant’s claimed combination of elements in claim 1 or method steps in claim 22 are not shown in any one embodiment (embodiments (1) through (3) as disclosed in the AAPA), the Examiner is not free to pick and chose elements from different embodiments and then combine them in the manner taught by the Appellant’s claims. It is incumbent upon the Examiner to establish some “reason” or “motivation” for picking and choosing elements from among the various prior art figures.

The Examiner's picking and choosing of elements from the three different embodiments of the AAPA ((1) Figures 3A and 3B, (2) Figures 4A and 4B, and (3) Figure 5) is clearly seen in the rejection of claims 1 and 22 on pages 4 & 5 of the Final Rejection. In the paragraph beginning "said address generating stage . . . ,", the Examiner references "ADD unit 170" from the second embodiment of Figures 4A and 4B with the "operands (x and y; see fig 3 B . . .) . . ." of the first embodiment of Figures 3A and 3B. Unbelievably, in response to Appellant's pointing out this inconsistency in the previously filed amendment (last paragraph page 18 of the Amendment filed July 7, 2006), the Examiner states that he is only relying upon embodiment 3 shown in Figure 5 ("Examiner based the rejection on the embodiment shown in figure 5." Final Rejection, page 14, section 32). The above noted quotes from the discussion on page 4 of the Final Rejection (regarding the Examiner's contention that the AAPA supports the claimed "address generation stage") confirms that the Examiner is relying upon bits and pieces combined from the three distinct embodiments discussed in the AAPA.

At best the Examiner may argue a §103 rejection based upon the three different embodiments in the AAPA, and, if so, the Examiner's failure to meet the requirement of providing some "reason" or "motivation" for combining different portions of the three different embodiments in the AAPA is simply not supported and the burden of proving the obviousness of a combination of elements has not been met.

D. The Examiner fails to identify where the steps of claim 22 are set out in AAPA

While the above distinctions of apparatus claim 1 also apply to independent method claim 22 (the Examiner states in section 8 on page 5 that “claim 22 recites equivalent limitations as claim 1 and is rejected under the same grounds”) they are herein incorporated by reference.

However, the Examiner’s failure to address the individually recited method steps is fatal to the rejection of claim 22 and claims 23-42 dependent thereon. For example, in the final rejection, nowhere is there discussed where any one embodiment of the AAPA’s disclosed three different embodiments teaches all five of the recited steps of “receiving,” “generating,” “adding together,” “routing” and “routing.” One example is in section 8 of the Final Rejection, where the Examiner states that “it is possible for multiplexer 130 to produce/generate the shifted value” While anything is possible, the test under §102 is whether the AAPA teaches that specifically claimed “generating” step in claim 22. The Examiner fails to identify any such teaching in the AAPA.

In fact, there is no specific analysis of the recited steps in claim 22 or indication where each of the steps is disclosed in one of the embodiments of the AAPA. The Examiner has simply not met his burden of proof.

**E. There is no support for the rejection of claims 1-42
under 35 USC §102 as anticipated by Applicant Admitted
Prior Art**

Each of the above Sections A-D are herein incorporated by reference. Each of these sections identifies a basis for traversal of the anticipation rejection of independent claims 1 and 22 and claims 2-21 and 23-42 dependent thereon. Appellant, instead of restating of these sections, will merely refer to the sections in the specific synopsis of the existing Final Rejection of claims 1-42 under §102.

Prior to any rejection of a claim, it is incumbent upon the Examiner to determine the scope of, and properly construe, the independent claims. Independent apparatus claim 1, as noted in Section A, is in means-plus-function format because no structure for performing the recited logic functions has been recited in the claim. As noted in Section D, the Examiner fails to specifically identify any disclosure of the specific steps set forth in method claim 22.

Because the Examiner has failed to properly construe independent claims 1 and 22, his general allegation that the claimed structures and the claimed method steps are anticipated in the Background of Invention portion of Appellant's specification is simply unsupported. Appellant has specifically identified in Sections B(1) and B(2) above that none of the three embodiments in the AAPA disclose the recited "address generation logic" or the "operand routing logic" of claim 1 and in Section D, the plurality of steps recited in claim 22. Because the Examiner has failed to identify where the AAPA discloses each claimed element

or method step and each claimed interrelationship, the rejection fails the *Lindemann* test applying 35 USC §102. The Examiner does not indicate where each of the recited logic structures or each of the recited claim 22 method steps is contained or disclosed in Appellant's Background.

Additionally, it is incumbent upon the Examiner to establish that the cited prior art teaches the elements "arranged as in the claim." *Id.* The Examiner does not address the fact that even if the claimed logic structures and the claimed method steps were disclosed somewhere in the 4-1/2 page Background of the Invention.¹ There is simply no disclosure of any of the claimed "arrangement" of these elements or method steps "as in the claim." The Examiner's failure to identify the claimed arrangement between logic elements in claim 1 and method steps in claim 22 is fatal to the rejection under 35 USC §102.

As set out in Section C, while the present rejection is under §102, it is noted that even a rejection under §103 requires some reason or motivation for combining different portions of different embodiments of the AAPA in the manner of Appellant's claims. As noted in Section C, there are several different embodiments of systems disclosed, and the Examiner merely picks and chooses elements from the various embodiments.

¹ It is noted that while the Examiner relies upon pages 1-5, a good portion of page 5 is a Summary of the Invention and thus is clearly not Applicant Admitted Prior Art.

Appellant believes that the anticipation rejection is avoided simply because the Examiner fails to identify any embodiment in the AAPA which teaches the claimed arrangement of the elements and/or method steps. However, should the Board believe a rejection under §103 to be appropriate, it is also noted that the Examiner does not provide any “reason” or “motivation” for picking and choosing elements from the three different embodiments and then combining them in the manner of the claims.

In this regard, it is noted that a recent directive from the Deputy Commissioner for Patent Operations, Margaret A. Focarino, sent on May 3, 2007, establishes that “in formulating a rejection under 35 USC §103(a) based upon a combination of prior art elements, it remains necessary [for the examiner] to identify the reason why a person of ordinary skill in the art would have combined the prior art elements in the manner claimed.”

Here, the Examiner’s picking and choosing of elements and method steps from multiple embodiments of the AAPA and then combining them in a rejection under §103 is clearly improper, let alone the fact that he combines them in a rejection under 35 USC §102.

Thus, the Examiner has failed to set out a *prima facie* case of anticipation under 35 USC §102 for apparatus claims 1-21 or method claims 22-42 and any further rejection thereunder is respectfully traversed.

VIII. CONCLUSION

The Examiner has failed to properly construe the independent apparatus claim to require structures which provide the recited logic functions and the independent method claim to require the recited method steps. The Examiner misapplies Federal Circuit law regarding claim construction of “means-plus-function” claims and the lack of the word “means” is not dispositive that the claim is not to be construed in the manner of §112 (6th paragraph). Properly claimed structures in claim 1 and steps in claim 22 are not identified in any one of the three disclosed embodiments of the AAPA. Nowhere is there referenced in the AAPA a disclosure of the claimed “address generation logic” or the “operand routing logic” elements or their recited interrelationships. The Examiner provides no reason why one of ordinary skill would combine elements taken from different embodiments of AAPA, and certainly no reason why they would be combined in the manner of the independent claims.

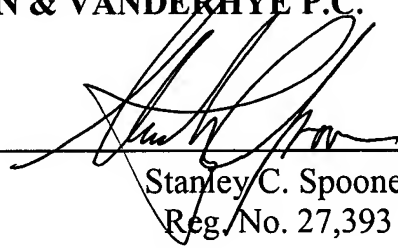
As a result of the above, there is simply no support for the rejection of Applicant's independent claim or claims dependent thereon under 35 USC §102. Thus, and in view of the above, the rejection of claims 1-42 under 35 USC §102 is clearly in error and reversal thereof by this Honorable Board is respectfully requested.

DIJKSTRA
Serial No. 10/633,362

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: _____

A handwritten signature in black ink, appearing to read 'Stanley C. Spooner', is written over a horizontal line. The signature is stylized with large, sweeping loops.

Stanley C. Spooner
Reg. No. 27,393

SCS:kmm
Enclosure



IX. CLAIMS APPENDIX

1. (previously presented) A data processing apparatus comprising:

a processor core operable to process a sequence of instructions, said processor core having a plurality of pipeline stages, one of said plurality of pipeline stages being an address generation stage operable to generate an address associated with an instruction for subsequent processing by said pipeline stages, said instruction being one from a first group of instructions or a second group of instructions, said address generation stage comprising:

address generation logic for receiving operands associated with said instruction, for generating a shifted operand from one of said operands, and for adding together, in dependence on said instruction, selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages; and

operand routing logic, in dependence on said instruction, for routing operands associated with instructions from said first group of instructions to said address generation logic and for routing operands associated with instructions from said second group of instructions via operand manipulation logic for manipulation of said operands prior to routing to said address generation logic.

2. (original) The data processing apparatus of claim 1, wherein said instruction relates to a memory access and said address indicates a location in memory to be accessed.

3. (original) The data processing apparatus of claim 1, wherein said first group of instructions comprises a first instruction which causes the processor core to logically add together two operands, and a second instruction which causes the processor core to logically add together one operand to another operand logically shifted by one of a predetermined number of bits.

4. (original) The data processing apparatus of claim 3, wherein said address generation logic is operable to generate said another operand logically shifted by one of a predetermined number of bits.

5. (original) The data processing apparatus of claim 3, wherein said second instruction causes the processor core to logically add together one operand to another operand logically shifted left by two bits.

6. (original) The data processing apparatus of claim 5, wherein said address generation logic is operable to generate said another operand logically shifted left by two bits.

7. (original) The data processing apparatus of claim 3, wherein said second instruction causes the processor core to logically add together one operand to another operand subject to only one preset logical shift operation.

8. (original) The data processing apparatus of claim 1, wherein said address generation logic is operable to perform only one predetermined logical shift operation and operands associated with all other logical shift operations required by instructions from said second group of instructions are routed via operand manipulation logic for manipulation of operands prior to routing to said address generation logic.

9. (original) The data processing apparatus of claim 3, wherein said second group of instructions comprises instructions which cause the processor core to logically add together one operand to another operand subject to any other logical shift operation.

10. (original) The data processing apparatus of claim 9, wherein said operand manipulation logic is operable, in dependence on said instruction, to generate said another operand logically shifted by any other number of bits.

11. (original) The data processing apparatus of claim 1, wherein said second group of instructions comprises instructions which cause the processor core to logically subtract one operand from another operand.

12. (original) The data processing apparatus of claim 11, wherein said operand manipulation logic is operable, in dependence on said instruction, to generate an inverse representation of one of said operand and said another operand.

13. (original) The data processing apparatus of claim 1, wherein said second group of instructions comprises a subtractive instruction for which said address is generated by subtracting a subtrahend operand from a minuend operand associated with said instruction, and said operand manipulation logic comprises subtraction operand generation logic operable to generate a negative representation of said subtrahend operand prior to routing to said address generation logic.

14. (original) The data processing apparatus of claim 1, wherein said address generation logic comprises:

operand generation logic operable to receive a first operand associated with said instruction and to generate a shifted operand representative of said first operand shifted by a predetermined number of bits;

operand selection logic operable, in dependence on said instruction, to select one of said first operand and said shifted operand as a selected operand; and

addition logic operable to add a second operand associated with said instruction to said selected operand to generate said address for subsequent processing by said pipelined stages.

15. (previously presented) The data processing apparatus of claim 14, wherein said first operand comprises 'n'-bits, where 'n' is a positive integer, said operand generation logic receives said first operand over an 'n'-bit input bus and provides said shifted operand on an 'n'-bit output bus, said operand generation logic comprising:

interconnection logic operable to couple lines of the 'n'-bit input bus with lines of the 'n'-bit output bus to perform a shift operation.

16. (original) The data processing apparatus of claim 14, wherein said operand selection logic is a two-input multiplexer.

17. (original) The data processing apparatus of claim 14, wherein said operand selection logic is operable to select one of said first operand and said shifted operand as a selected operand in response to a selection signal generated by instruction decoder logic.

18. (original) The data processing apparatus of claim 14, wherein said addition logic is a two-operand adder.

19. (original) The data processing apparatus of claim 1, wherein said operand routing logic is operable to route operands in response to a routing signal generated by instruction decoder logic.

20. (original) The data processing apparatus of claim 1, wherein said instruction is a subtraction instruction which causes the processor core to generate said address by subtracting a subtrahend operand in the form of an immediate from a minuend operand, and said data processing apparatus comprises instruction decoder logic operable to provide said subtrahend operand in negative form to said address generation stage and to generate a routing signal to cause said operand routing logic to route operands to said address generation logic.

21. (original) The data processing apparatus of claim 1, wherein said instruction is one of a load instruction and a store instruction.

22. (previously presented) In a data processing apparatus comprising a processor core operable to process a sequence of instructions, said processor core having a plurality of pipeline stages, one of said plurality of pipeline stages being an address generation stage operable to generate an address associated with an instruction for subsequent processing by said pipeline stages, said instruction being one from a first group of instructions or a second group of instructions, a method of generating said address comprising the steps of:

- a) receiving, at address generation logic, operands associated with said instruction;
 - b) generating a shifted operand from one of said operands;
 - c) adding together, in dependence on said instruction, selected ones of said operands and said shifted operand to generate said address for subsequent processing by said pipeline stages;
 - d) routing, in dependence on said instruction, operands associated with instructions from said first group of instructions to said address generation logic;
- and

e) routing, in dependence on said instruction, operands associated with instructions from said second group of instructions via operand manipulation logic for manipulation of said operands prior to routing to said address generation logic.

23. (original) The method of claim 22, wherein said instruction relates to a memory access and said address indicates a location in memory to be accessed.

24. (original) The method of claim 22, wherein said first group of instructions comprises a first instruction which causes the processor core to logically add together two operands, and a second instruction which causes the processor core to logically add together one operand to another operand logically shifted by one of a predetermined number of bits.

25. (original) The method of claim 24, wherein said step (b) comprises generating said another operand logically shifted by one of a predetermined number of bits.

26. (original) The method of claim 24, wherein said second instruction causes the processor core to logically add together one operand to another operand logically shifted left by two bits.

27. (original) The method of claim 26, wherein said step (b) comprises generating said another operand logically shifted left by two bits.

28. (original) The method of claim 24, wherein said second instruction causes the processor core to logically add together one operand to another operand subject to only one preset logical shift operation.

29. (original) The method of claim 22, wherein said address generation logic is operable to perform only one predetermined logical shift operation.

30. (original) The method of claim 24, wherein said second group of instructions comprise instructions which cause the processor core to logically add together one operand to another operand subject to any other logical shift operation.

31. (original) The method of claim 30, wherein said operand manipulation logic is operable, in dependence on said instruction, to generate said another operand logically shifted by any other number of bits.

32. (original) The method of claim 22, wherein said second group of instructions comprises instructions which cause the processor core to logically subtract one operand from another operand.

33. (original) The method of claim 32, wherein said operand manipulation logic is operable, in dependence on said instruction, to generate an inverse representation of one of said operand and said another operand.

34. (original) The method of claim 22, wherein said second group of instructions comprises a subtractive instruction for which said address is generated by subtracting a subtrahend operand from a minuend operand associated with said instruction, and said operand manipulation logic comprises subtraction operand generation logic operable to generate a negative representation of said subtrahend operand prior to routing to said address generation logic.

35. (previously presented) The method of claim 22, wherein said step (a) comprises the step of receiving a first operand associated with said instruction, said step (b) comprises the step of generating a shifted operand representative of said first operand shifted by a predetermined number of bits and said step (c) comprises the steps of: selecting one of said first operand and said shifted operand as a selected operand; and adding a second operand associated with said

instruction to said selected operand to generate said address for subsequent processing by said pipelined stages.

36. (previously presented) The method of claim 35, wherein said first operand comprises 'n'-bits, where 'n' is a positive integer, said step (a) comprises receiving said first operand over an 'n'-bit input bus and said step (b) comprises providing said shifted operand on an 'n'-bit output bus by providing interconnection logic operable to couple lines of the 'n'-bit input bus with lines of the 'n'-bit output bus to perform a shift operation.

37. (original) The method of claim 35, wherein said selecting step is performed by a two-input multiplexer.

38. (original) The method of claim 35, wherein said selecting step comprises selecting one of said first operand and said shifted operand as a selected operand in response to a selection signal generated by instruction decoder logic.

39. (original) The method of claim 35, wherein said addition step is performed by a two-operand adder.

40. (original) The method of claim 22, wherein said steps (d) and (e) comprise routing operands in response to a routing signal generated by instruction decoder logic.

41. (original) The method of claim 22, wherein said instruction is a subtraction instruction which causes the processor core to generate said address by subtracting a subtrahend operand in the form of an immediate from a minuend operand, and said data processing apparatus comprises instruction decoder logic operable to provide said subtrahend operand in negative form to said address generation stage and to generate a routing signal to cause said operands to be routed to said address generation logic.

42. (original) The method of claim 22, wherein said instruction is one of a load instruction and a store instruction.

X. EVIDENCE APPENDIX

1. Webster's Third New International Dictionary, page 1330.

DIJKSTRA
Serial No. 10/633,362

XI. RELATED PROCEEDINGS APPENDIX

None.